**C# ADO.NET Architecture**

**ADO.NET**



**ADO.NET** is a data access technology from Microsoft [.Net Framework](http://vb.net-informations.com/framework/what_is_net_framework.htm) , which provides communication between relational and non-relational systems through a common set of components .**ADO.NET** consist of a set of Objects that expose data access services to the .NET environment. ADO.NET is designed to be easy to use, and Visual Studio provides several wizards and other features that you can use to generate **ADO.NET** data access code.

**Data Providers and DataSet**



The two key components of ADO.NET are **Data Providers** and**DataSet** . The .Net Framework includes mainly three Data Providers for ADO.NET. They are the Microsoft **SQL Server Data Provider** , **OLEDB Data Provider** and **ODBC Data Provider** . SQL Server uses the SqlConnection object , OLEDB uses the OleDbConnection Object and ODBC uses OdbcConnection Object respectively.

[C# SQL Server Connection](http://csharp.net-informations.com/data-providers/csharp-sql-server-connection.htm)

[C# OLEDB Connection](http://csharp.net-informations.com/data-providers/csharp-oledb-connection.htm)

[C# ODBC Connection](http://csharp.net-informations.com/data-providers/csharp-odbc-connection.htm)



The four Objects from the [.Net Framework](http://vb.net-informations.com/framework/what_is_net_framework.htm) provides the functionality of Data Providers in the ADO.NET. They are**Connection** Object, **Command** Object , **DataReader** Object and**DataAdapter** Object. The Connection Object provides physical connection to the Data Source. The Command Object uses to perform SQL statement or stored procedure to be executed at the Data Source. The DataReader Object is a stream-based , forward-only, read-only retrieval of query results from the Data Source, which do not update the data. Finally the DataAdapter Object , which populate a Dataset Object with results from a Data Source .

**DataSet**



**DataSet** provides a disconnected representation of result sets from the Data Source, and it is completely independent from the Data Source. DataSet provides much greater flexibility when dealing with related Result Sets.

**DataSet** consists of a collection of **DataTable** objects that you can relate to each other with DataRelation objects. The DataTable contains a collection of **DataRow** and **DataCoulumn** Object which contains Data. The DataAdapter Object provides a bridge between the DataSet and the Data Source. From the following section you can see each of the ADO.NET components in details with **C# Source Code** .

**Advantages of ADO.Net over ADO**

ADO stands for ActiveX Data Objects and it relies on COM whereas ADO.NET relies on managed providers defined by the .NET CLR (Common Language Runtime). ADO.NET provides consistent access to data sources such as SQL Server, as well as data sources exposed through OLE DB and XML. While there are similarities between ADO and ADO.NET, the way they operate and their foundations are quite different. The following are some Advantages of ADO.Net over ADO in basic level.

A major difference in creating connections with ADO and ADO.NET is that ADO fits all connections to all types of data sources into a single Connection object. ADO.NET can have separate Objects that represent connections to different data sources. In ADO.NET you can create multiple data provider namespaces to connect specifically with a particular data source, making access faster and more efficient and allowing each namespace to exploit the features of its targeted data provider.

 **SqlConnection connection**

 **connection = new SqlConnection("connetionString");**

 **OleDbConnection connection;**

 **connection = new OleDbConnection("connetionString");**

ADO allows you to create client side cursors only whereas ADO.NET gives you the choice of either using client side or server side cursors.

ADO.NET introduces a new way of getting a single value from a query's results when you expect only one row and one column to return. The ADO.NET command object has an ExecuteScalar method which returns the first row and column's value from its associated query.

ADO.Net dataset represents in memory representation of a database. ADO recordsets is merely a set of rows retrieved from a data source.

ADO recordsets can hold data from one data source at a time. ADO.Net datasets can hold data from various sources and integrate the data and write it back to one or several data sources.

The ADO.NET Framework supports two models of Data Access Architecture, Connection Oriented Data Access Architecture and Disconnected Data Access Architecture.

In the case of Data Communication , ADO objects communicate in binary mode while ADO.NET uses XML for passing the data.